ARMY RESEARCH LABORATORY

Image-based Localization in Urban Environments

by Philip David

ARL-MR-0738 **March 2010**

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory
Adelphi, MD 20783-1197

ARL-MR-0738 **March 2010**

Image-based Localization in Urban Environments

Philip David
Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
March 2010	DRI	

4. TITLE AND SUBTITLE
Image-based Localization in Urban Environments

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Philip David

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
U.S. Army Research Laboratory
ATTN: RDRL-CII-A
2800 Powder Mill Road
Adelphi, MD 20783-1197

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-MR-0738

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report describes an efficient algorithm to accurately determine the position and orientation of a camera in an outdoor urban environment using camera imagery acquired from a *single* location on the ground. The requirement to operate using imagery from a single location allows a system using our algorithms to generate instant position estimates and ensures that the approach may be applied to both mobile and immobile ground sensors. Localization is accomplished by registering visible ground images to urban terrain models that are easily generated offline from aerial imagery. Provided there are a sufficient number of buildings in view of the sensor, our approach provides accurate position and orientation estimates, with position estimates that are more accurate than those typically produced by a global positioning system (GPS).

15. SUBJECT TERMS
localization, aerial to ground image registration, omnidirectional camera, vanishing points

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Philip David
Unclassified	Unclassified	Unclassified	UU	28	19b. TELEPHONE NUMBER (*Include area code*) (301) 394-5603

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

Acknowledgments

I would like to thank Mr. Nick Fung of the U.S. Army Research Laboratory (RDRL-CII-A) for providing the panoramic imagery and assisting with experiments.

INTENTIONALLY LEFT BLANK.

1. Objective

Imaging sensors are currently being deployed in large numbers on vehicle systems and will likely be deployed in the future on individual Soldiers as well. These sensors are intended to serve many purposes, including target detection and tracking, detection and location of hostile fire, navigational aid, and terrain model acquisition. Information regarding events observed on the battlefield is most useful when these events can be accurately localized with respect to some larger coordinate system. Localization allows multiple, non-collocated systems to exchange and fuse information, and coordinate their actions. The first step to localizing observed events is to locate the position and orientation* of the observing sensor. There are many ways to determine the position of a sensor in an environment. These include using a global positioning system (GPS), compass, TV or cell phone networks, and landmark recognition from optical or range data. None of these methods will solve the localization problem all of the time: GPS works best in unobscured outdoor environments, but does not provide orientation and is not accurate enough for some applications (such as autonomous navigation of unmanned ground vehicles). Cell phone networks can provide indoor and outdoor localization, but are accurate to only 100 m. Landmark recognition can give accurate position and orientation, but may be unreliable and computationally intense, and require laborious offline terrain modeling. Some combination of these techniques will be needed to robustly solve the battlefield sensor localization problem. In the short term, a system is envisioned that performs localization using a combination of GPS and landmark recognition. GPS, when available, will provide the rough positioning and the landmark recognition subsystem will refine that position and fill in the gaps during GPS outages. The research described here is focused on localization using visual landmark recognition in urban environments.

The goal of our research is to develop and demonstrate efficient and accurate algorithms to determine the position and orientation of a camera in an outdoor urban environment using camera imagery acquired from a *single* location on the ground. The requirement to operate using imagery from a single location allows a system using our algorithms to generate instant position estimates and ensures that the approach may be applied to both mobile and immobile ground sensors. Localization is accomplished by registering visible ground images to urban terrain models that are easily generated offline from aerial imagery. Provided there are a sufficient number of buildings in view of the sensor, our approach provides accurate position and orientation estimates, with position estimates that are more accurate than those typically produced by GPS.

*Hereafter, for conciseness, the terms *position* and *location* will often be used synonymously with the phrase *position and orientation*.

2. Approach

The position of a camera can be determined from objects observed in the scene by recognizing landmarks (i.e., immobile objects like buildings, statues, natural features, etc.) and retrieving their prerecorded positions from an existing database or terrain model. Approaches to landmark recognition may be broadly classified as either appearance-based or model-based. Appearance-based approaches represent objects as collections of two-dimensional (2-D) images. Model-based approaches represent objects in terms of some higher level structures (e.g., 2-D or three-dimensional [3-D] geometric models). In both approaches, a search is performed to match a new image to the model. Although research in visual landmark recognition has been ongoing for over 30 years, all existing techniques seem to suffer from at least one of the following problems (*2, 3*): (1) The sensor must have previously observed the scene from similar vantage points for it to be recognized; (2) excessive computations are needed to match image data to a terrain model, thus limiting real-time operation; (3) accurate localization information cannot be obtained from recognized landmarks; and (4) background clutter causes recognition and localization errors.

Our approach to landmark recognition addresses all of these issues. We register ground-based imagery to a terrain model that is easily created from a single aerial image. Our terrain model consists of a 2-D map of building footprints. Thus, the sensor does not have to visit an area of the terrain previously to localize itself in that terrain. The location of the camera in the urban terrain is determined by estimating, from a single image, the footprints of visible building facades and then registering this local footprint to the terrain model. Both the local footprint estimation and the registration steps are fast. Local building footprint estimation is performed using image vanishing points to compute the 3-space orientations on the ground plane of line segments detected in an omnidirectional camera image. In other words, information derived from vanishing points is used to identify image line segments that correspond to vertical building facades and then used again to project these line segments onto the ground plane. Given the ground plane projections, a vector describing the footprint orientations at equal angles over a 360° field of regard is computed. The local footprint orientation (LFO) vector is then matched to the 2-D terrain model to determine the camera's position and orientation. Each of these steps is described in more detail in sections 2.1 through 2.4.

2.1 Sensors for Air-to-Ground Registration

As mentioned previously, our terrain model consists of a 2-D map of building footprints. The footprint of a building consists of the projection onto a horizontal plane of all large vertical facades of that building. The building footprints for an urban environment can be created in a number of ways. Numerous approaches for automated building footprint detection exist using high-resolution aerial monoscopic visible (*0, 0*), stereoscopic visible (*0–0*), and Light Detection and Ranging (LIDAR) (*0–0*) data. Approaches using LIDAR are currently more robust than

those using only optical imagery. In fact, there is a commercial product named LIDAR Analyst™ (*0*) that automatically extracts building footprints and 3-D computer-aided design (CAD) models from high-resolution LIDAR data. Figure 1 shows some of the outputs of this product found on the company's Web site. As shown in this figure, it is possible to automatically generate accurate building footprint models using existing systems. However, because the focus of this research is on sensor localization and not terrain modeling, we did not use any of these existing methods to generate our models, but instead generated our building footprint models by hand from single aerial images. This was a cheap and fast way for us to test our method. A graphical user interface was written that allows the user to easily and quickly draw lines on top of an image and save these lines as a terrain model. Figure 2 shows an example of a publically available (from the U.S. Geological Survey) aerial image of part of the U.S. Army Research Laboratory (ARL) and the building footprint model that we manually generated from that image.

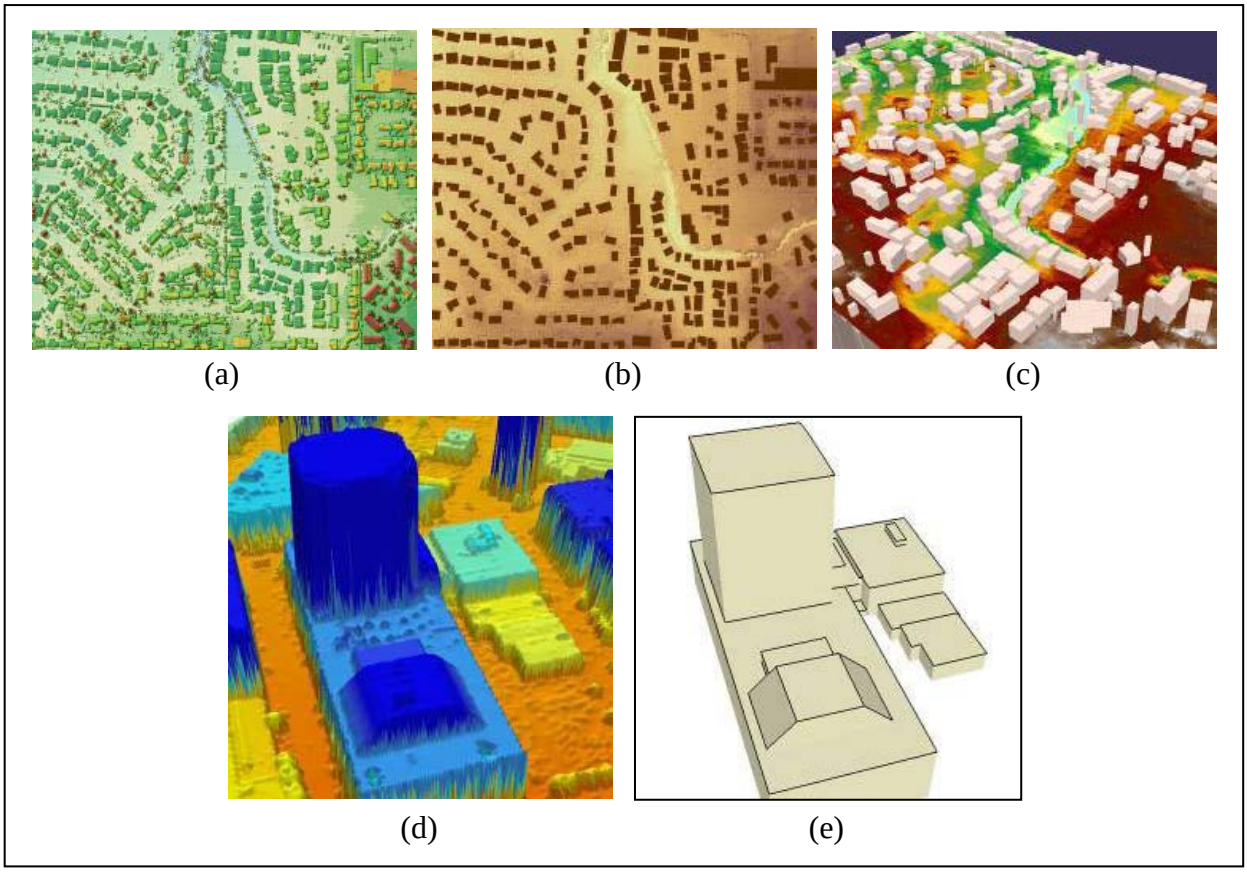

Figure 1. Visual Learning Systems, Inc., LIDAR Analyst system: (a) a LIDAR digital elevation model (DEM) (i.e., raw LIDAR data), (b) building footprints extracted from the LIDAR DEM, (c) 3-D building models generated from DEM, (d) a close-up of a LIDAR DEM of a complex building, and (e) a 3-D model generated from DEM of the complex building.

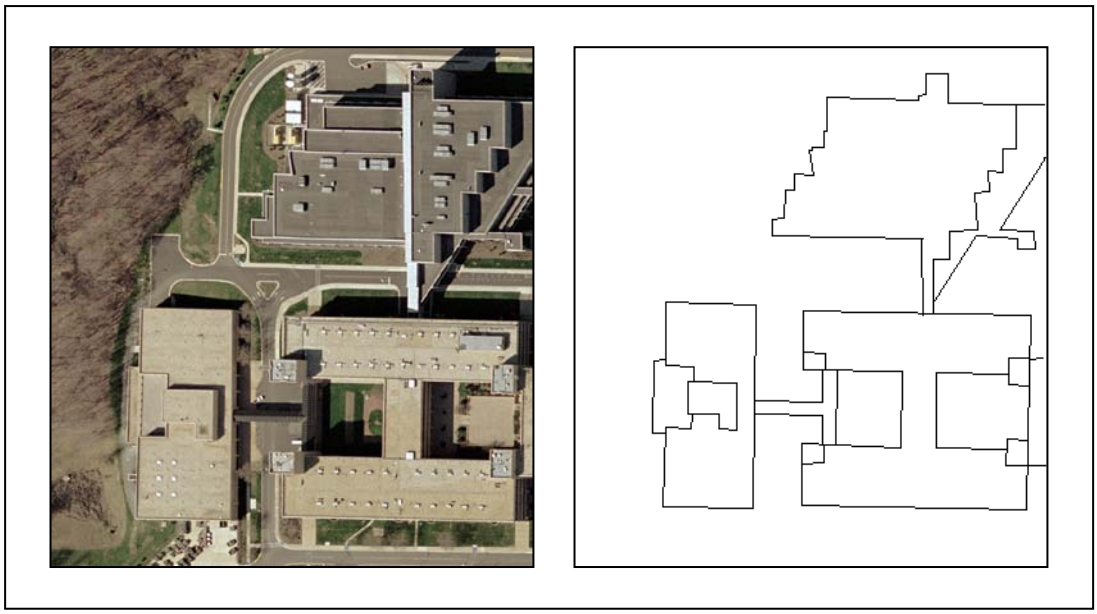

Figure 2. Aerial image of ARL (left) and the building footprint model manually generated from that image (right).

The system to be localized must possess a sensor that will observe the surrounding environment. Because LFO vectors describe building footprints over the full 360° range of viewing, any sensor for this purpose that does not have a 360° field of view would need to be panned to cover the full range. A number of different sensors may be used, including a monocular visible camera, a stereo visible camera pair, or a LIDAR camera. As stereo and LIDAR cameras generate range directly, they should allow for faster and more accurate calculation of the LFO vectors. However, these two sensors are usually only accurate at short ranges. The monocular visible camera has the ability to observe over much longer ranges, but requires significantly more processing to generate LFO vectors. In our research, we use the Point Grey Research, Inc., Ladybug®2 spherical camera shown in figure 3. This digital video camera system consists of six 0.8 megapixel color charge-coupled device (CCD) image sensors, each with a 2.5-mm focal length lens, integrated into a single enclosure. Five of the cameras are positioned in a horizontal ring and one is positioned vertically. This enables the camera to collect imagery from more than 75% of the full viewing sphere. An integrated 12-bit analog-to-digital converter along with an IEEE-1394b (FireWire) interface is used to stream full 12 megapixel images at 15 FPS to the host system. This camera system is small (110 mm x 100 mm x 141 mm, L×W×H) and light enough (1190 g) to be mounted on a portable robot. Figure 4 shows a set of six images that were simultaneously acquired from the Ladybug camera as it sat on the ground in an outdoor courtyard at ARL.

Figure 3. Point Grey Research Ladybug2 spherical vision camera system.

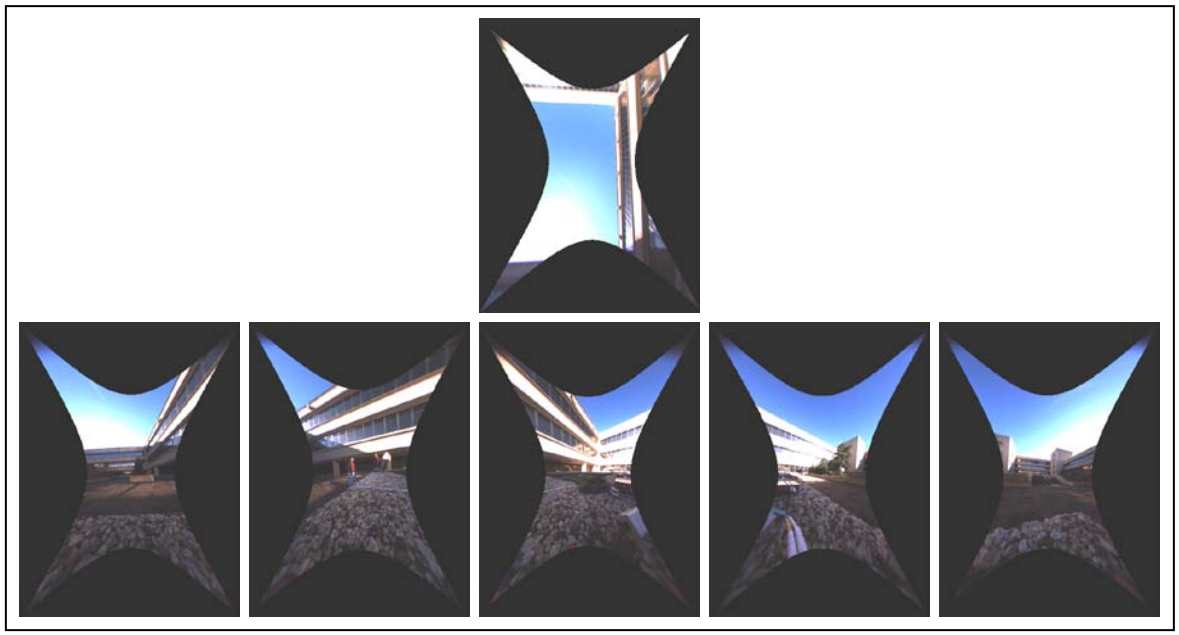

Figure 4. Images from the Ladybug camera taken inside the ARL courtyard. The top image is from the vertical camera and the lower five images are from cameras in the horizontal ring. The pincushion distortion effect apparent in these images is a result of warping of the original images (not shown) to correct the optical distortions that were present in the original images.

2.2 Image Vanishing Points

Optical image formation is usually modeled using perspective projection. In an idealized perspective camera, a point in 3-space is mapped onto an image at the location where a ray connecting the center of projection (the lens center) and the 3-space point intersects the image. Optical aberrations such as focus and lens distortion cause deviations from this model, but these effects can be accounted for via standard camera calibration techniques. The planar perspective image **x** of a 3-space point **X** can be modeled as

$$\mathbf{x} = \mathrm{K}\left[\mathrm{I} \mid \mathbf{0}\right]\mathbf{X}, \tag{1}$$

where **x** and **X** are the homogeneous representations of the image and 3-space points†, respectively; I is the 3x3 identity matrix; **0** is the length three column zero vector; and K is the camera calibration matrix.

$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{2}$$

Equation 1 makes the assumption that the 3-space coordinates of **X** are given in the camera frame of reference. The parameters of the camera calibration matrix are α_x and α_y, the focal lengths in the horizontal and vertical directions, respectively; *s*, the axis skew; and x_0 and y_0, the position of the optical axis on the image plane (*1*).

Under perspective projection, an infinitely long line in the world can have a finite extent in the image; the image of the point at infinity on this line is called the line's vanishing point. Parallel lines in the world that are not parallel to the image plane will be imaged as converging lines that intersect at a single finite vanishing point. When image points and lines are represented using homogeneous coordinates, the image of any set of parallel world lines, whether or not they are parallel to the image plane, will intersect at a common vanishing point. This point may be a point at infinity, but these points are treated identically to finite vanishing points. Figure 5 shows the image of an urban environment with some parallel world lines and their vanishing points identified.

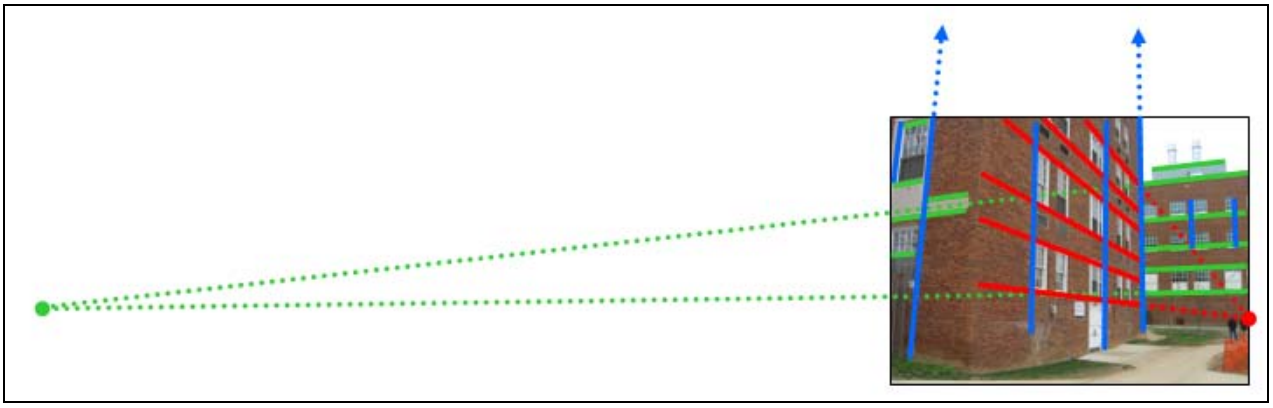

Figure 5. The image of parallel 3-space world lines intersecting at a common vanishing point in the image. Vanishing points (shown as colored dots in this figure) may lie inside or outside the image. The blue lines intersect at a point high above the image.

†Homogeneous coordinates are used throughout this report to represent image and 3-space points.

The 3-space direction of a line relative to the camera reference frame may be determined from the line's vanishing point as follows. The line through the 3-space point **A** with direction $\mathbf{D} = (\mathbf{d}^{\mathrm{T}}, 0)^{\mathrm{T}}$ ($\mathbf{d} \in R^3, \mathbf{d} \neq \mathbf{0}$) may be represented as the set of points with homogeneous coordinates $\mathbf{X}(\lambda) = \mathbf{A} + \lambda\mathbf{D}$. From equation 1, the image of point $\mathbf{X}(\lambda)$ is

$$\mathbf{x}(\lambda) = \mathrm{K}[\mathrm{I} \mid \mathbf{0}]\mathbf{A} + \lambda\mathrm{K}[\mathrm{I} \mid \mathbf{0}](\mathbf{d}^{\mathrm{T}}, 0)^{\mathrm{T}} = \mathbf{a} + \lambda\mathrm{K}\mathbf{d}\,, \tag{3}$$

where **a** is the image of the point **A**. The vanishing point **v** of the line is then

$$\mathbf{v} = \lim_{\lambda\to\infty} \mathbf{x}(\lambda) = \lim_{\lambda\to\infty} (\mathbf{a} + \lambda\mathrm{K}\mathbf{d}) = \mathrm{K}\mathbf{d}\,. \tag{4}$$

Thus, given the vanishing **v** point of a 3-space line and the camera calibration matrix K, the 3-space direction of the line is

$$\mathbf{d} = \mathrm{K}^{-1}\mathbf{v} \tag{5}$$

and the projection onto a ground plane with unit normal $\mathbf{n} \in R^3$ is

$$\mathbf{d}' = \mathrm{K}^{-1}(\mathbf{v} - (\mathbf{v}\cdot\mathbf{n})\mathbf{n}). \tag{6}$$

2.3 Estimating the 3-space Orientation of Image Lines

The first step in our approach to estimating the 3-space orientations of image line segments is to detect the vanishing points in the image. Vanishing points are detected as follows. The Canny edge detector (*14*) with hysteresis thresholding is first applied to generate a binary image of the edge points (figure 6b). Straight line segments are then extracted from this edge image by linking edges into contours (figure 6c) and then splitting the contours into straight segments (*15*). The final line segments are those whose sum of squared distances to the contour points are minimized and whose length is at least 5 pixels long (figure 6d).

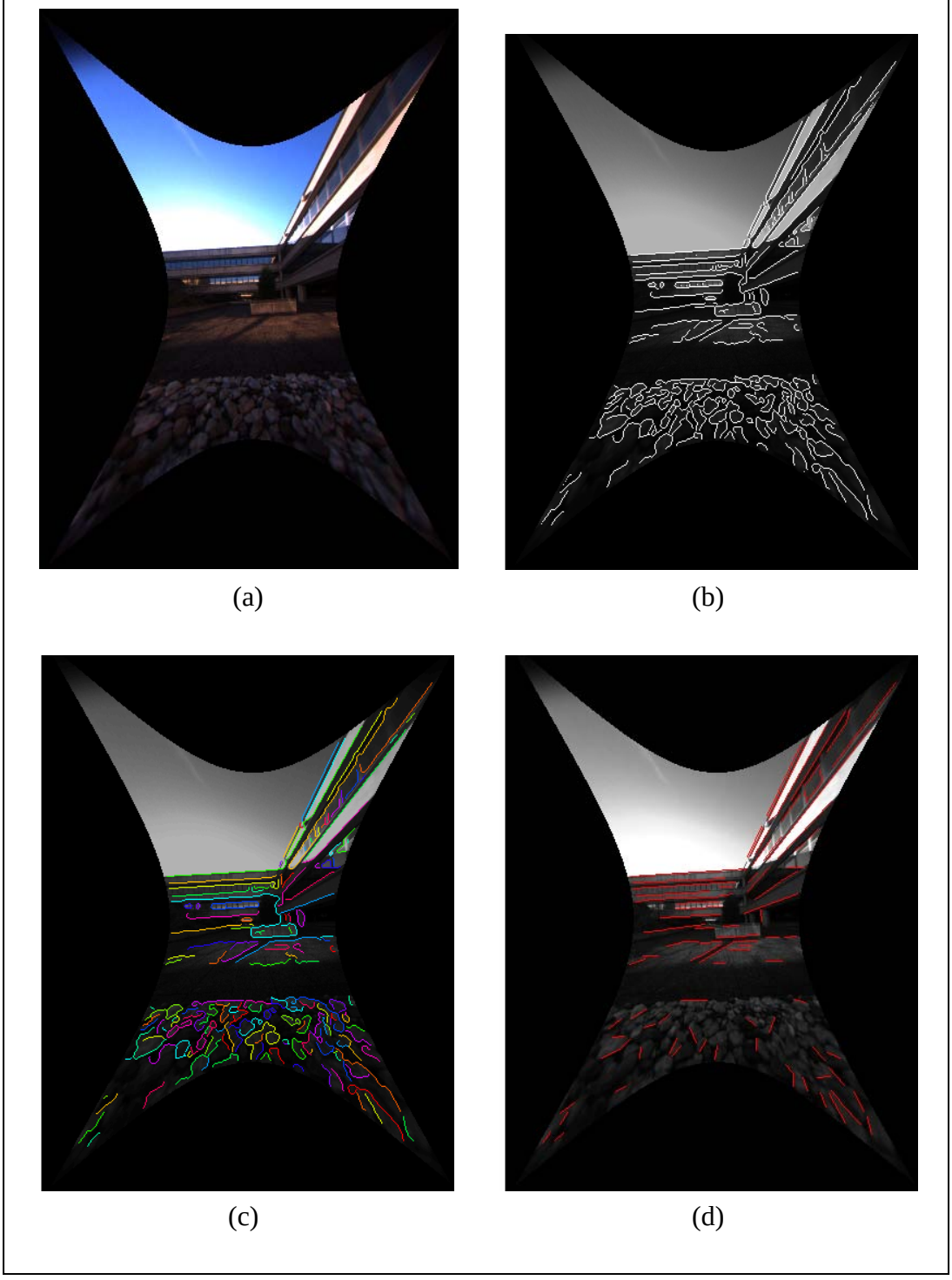

Figure 6. Image processing for straight line segment detection: (a) original image, (b) Canny edges, (c) edge contours, and (d) straight line segments fit to contours.

Each line segment L_i is identified by its two endpoints: $L_i = \left\{\left(x_i^1, y_i^1\right), \left(x_i^2, y_i^2\right)\right\}$. For efficiency in computing the image vanishing points, for each line segment L_i we compute the normalized homogeneous representation of the coincident infinite line, l_i; the endpoints, $\mathbf{e}_i^1$ and $\mathbf{e}_i^2$; and the midpoint, $\mathbf{m}_i$. These are calculated according to

$$\begin{aligned}
\mathbf{e}_i^1 &= \left(x_i^1, y_i^1, 1\right)^{\mathrm{T}}, \\
\mathbf{e}_i^2 &= \left(x_i^2, y_i^2, 1\right)^{\mathrm{T}}, \\
\mathbf{m}_i &= \left(\left(x_i^1 + x_i^2\right)/2, \left(y_i^1 + y_i^2\right)/2, 1\right)^{\mathrm{T}}, \\
\mathbf{l}_i' &= \mathbf{e}_i^1 \times \mathbf{e}_i^2, \\
\mathbf{l}_i &= \mathbf{l}_i' \Big/ \sqrt{\mathbf{l}_i'(1)^2 + \mathbf{l}_i'(2)^2}.
\end{aligned} \tag{7}$$

The Random Sample Consensus (RANSAC) algorithm (*16*) is then applied several times to the above data; each trial is used to locate the single vanishing point with the most support. Before each new trial, the data supporting the vanishing point found in the previous trail are removed. This process is repeated until V_{max} = 4 vanishing points are found, or until the size of the largest consensus set is less than $S_{\min}$ = 20. On each trial of RANSAC, T=50, random samples of line pairs are examined. The line pair $\mathbf{l}_i$ and $\mathbf{l}_j$ seeds a potential vanishing point $\mathbf{v}_{ij}$ when the line segments L_i and L_j are each at least H_{seed} = 15 pixels long and when their angle is no longer than $\Theta_{\text{seed}} = 40^\circ$. The initial vanishing point of the line pair is $\mathbf{v}_{ij} = \mathbf{l}_i \times \mathbf{l}_j$. The normalized line through $\mathbf{v}_{ij}$ and the midpoint of line segment L_k is given by $\mathbf{l}_{ijk} = \mathbf{l}_{ijk}' \Big/ \sqrt{\mathbf{l}_{ijk}'(1)^2 + \mathbf{l}_{ijk}'(2)^2}$, where $\mathbf{l}_{ijk}' = \mathbf{v}_{ij} \times \mathbf{m}_k$. Then, line segment L_k is considered to support $\mathbf{v}_{ij}$ and is added to the consensus set C_{ij} when the perpendicular distance, $d_{ijk} = \mathbf{l}_{ijk} \times \mathbf{e}_k^1$, from one endpoint $\mathbf{e}_k^1$ of L_k to $\mathbf{l}_{ijk}$ is no larger than $D_{\text{sup}} = 3$ pixels and when the angle between these lines is no larger than $\Theta_{\text{sup}} = 3^\circ$. All line segments in the largest consensus set are used to estimate the final location of the vanishing point, $\mathbf{v}^*$. $\mathbf{v}^*$ is required to minimize the weighted sum, for all lines L_t in the consensus set, of the squared distance of line segment end points to the line through $\mathbf{v}^*$ and $\mathbf{m}_t$. $\mathbf{v}^*$ is found using standard nonlinear optimization routines. Figure 7 shows the results of using this algorithm to classify the 3-space orientation of the line segments detected in the image of figure 6a. Figure 8 shows the same results for the six Ladybug camera images shown in figure 4

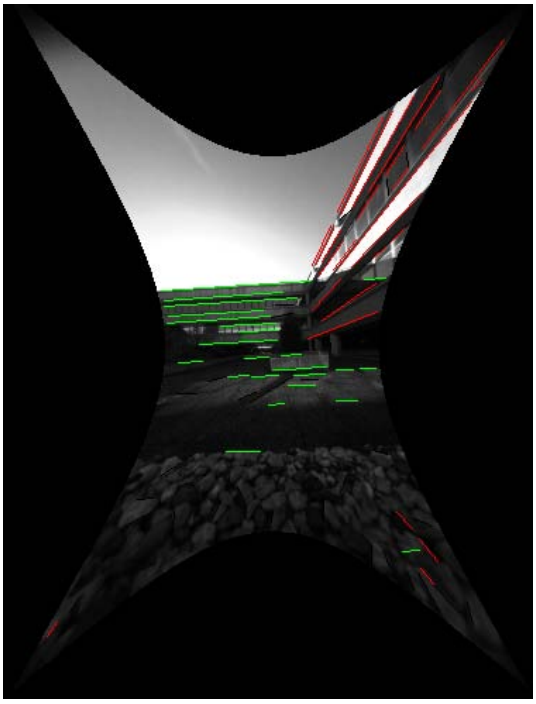

Figure 7. Results of using the vanishing point algorithm to classify the 3-space orientation of the line segments detected in the image of figure 6a. All lines of the same color have the same 3-space orientation.

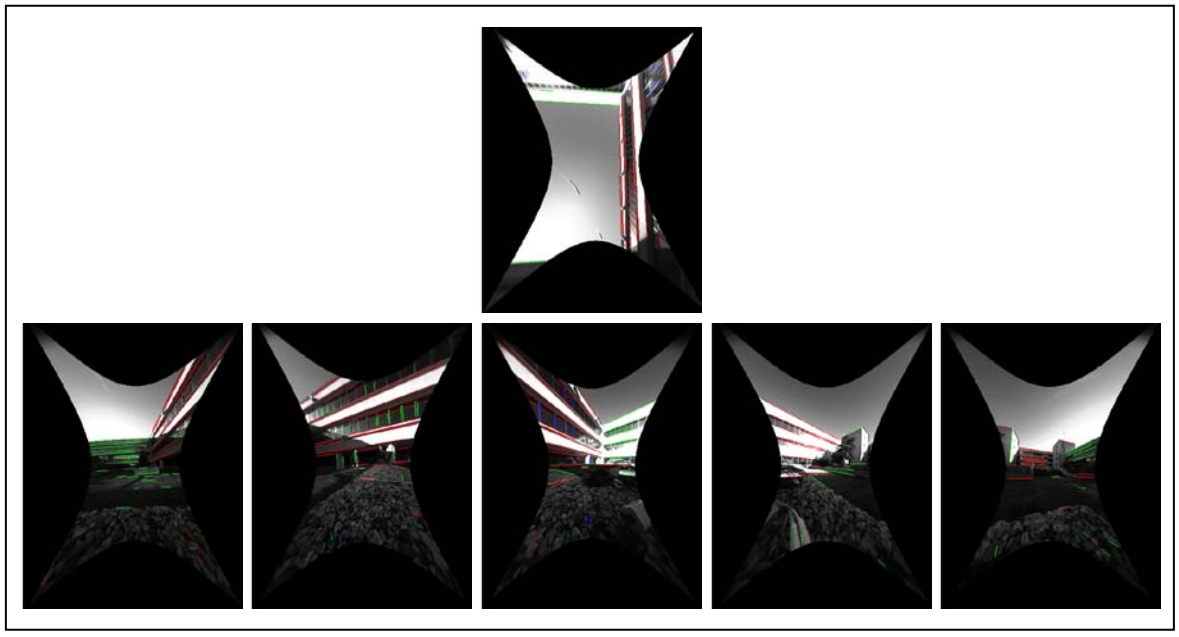

Figure 8. Line segments are detected in the ARL courtyard images from figure 4 and are color coded (independently in each image) based on the estimated orientation of the associated 3-space line.

In a typical urban environment, building facades are planar and orthogonal to the ground plane. Furthermore, markings on most building facades consist of two sets of orthogonal 3-space lines, one set that is orthogonal to the ground plane and one set that is parallel to the ground plane. The vanishing point of the set that is orthogonal to the ground plane determines the vertical orientation of the camera relative to the ground plane. The vanishing points of 3-space lines that are parallel to the ground plane determine the orientations of the respective façades when projected down onto the ground plane. An omnidirectional camera (the Ladybug) is used to ensure that the vertical vanishing point will be detected. This is essential in order to determine the orientation of the ground plane and, from this, the orientation of building facades. The vertical vanishing point is identified as the vanishing point whose position is nearest to the center of the overhead image from the set of six Ladybug images. This vanishing point defines the ground plane normal, **n**. Given **n**, the orientation of the projection onto the ground plane of any classified line segment is computed according to equation 6.

2.4 Position Estimation

For a calibrated camera (i.e., a camera where the camera calibration matrix in equation 2 is known), every pixel in the image corresponds to a specific horizontal and vertical viewing angle. All image line segments whose ground plane orientation was determined, as described in the previous subsection, are used to estimate the local building footprint. For each horizontal viewing angle, the orientation of the building footprint in that direction is assigned the dominant orientation of all classified line segments in that direction. The LFO vector consists of the dominant orientation for 36 directions equally spaced over the 360° viewing plane. For each $\theta \in \{0^\circ, 10^\circ, 20^\circ, \ldots, 380^\circ\}$, the average is computed over all viewing directions in the range $[\theta - 5^\circ, \theta + 5^\circ]$.

Given the position of a camera with respect to the building footprint terrain model, the LFO vector is computed similarly to the process described in the previous paragraph, except the terrain model is used to compute the ground plane orientations instead of the image line segments. If the ray from the given position and in the specific viewing direction intersects a building footprint, then this angle (which is in the range 0 to 180°) defines the building footprint orientation in that viewing direction. If the ray does not intersect and building footprint, then a value of 0 is assigned. This process is illustrated in figure 9 for a region of the building footprint map illustrated in figure 2.

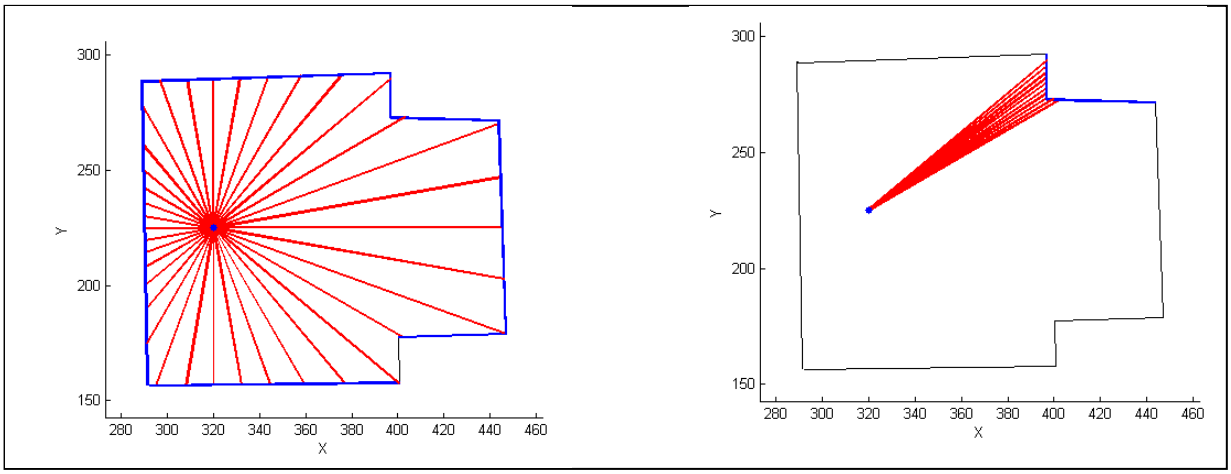

Figure 9. Computation of the LFO vector for the location in the building footprint terrain model marked with the blue dot. Thirty-six footprint orientations are assigned from 36 equally spaced viewing directions (left). For each of these viewing directions, the ground plane orientation is computed as the average over all 3-space line orientations that are intersected by a ray within 5° of the given viewing direction. One such viewing direction is shown in the right image.

To determine the position and orientation of a camera from its omnidirectional images, we first process the images as described previously to generate the camera's LFO vector. This vector is then matched to those in the building footprint terrain model using a gradient descent or some other similar optimization scheme. The estimated location of the camera may be used, if available, to initialize this search.

3. Results

Because software has not been completed to integrate all components of our algorithm, we evaluated the approach using a simulation. We assumed that, at any location in the terrain model, our image processing algorithms were able to correctly estimate, to a small error, 90% of the 36 elements of the LFO vector. That is, 10% of the 36 elements of any LFO vector were assigned random values ranging from 0 to 180°. Furthermore, we assumed that the correctly estimated values had errors that were normally distributed with a mean of 0° and a standard deviation of 5°. Figures 10 and 11 show the true and estimate location of a camera for two different trials. In all experiments, the localization error was less than 0.5 m. The color at any point in these figures is proportional to the difference between the estimated LFO vector and the LFO vector at that point. This gives an indication of shape of the error surface and shows how close the initial guess must be to the true position in order for the algorithm to find an answer that is close to the true position of the camera. It can be seen that the error surfaces are smooth with fairly large basins of attraction. Thus, the local optimization algorithm usually found very good solutions. Note also, that the global minimum (over the entire region of the terrain model) is always very close to the true position of the camera (this was the case for all experiments that

we ran with the above-given parameters). Because the search space was at most 3-D (two dimensions for *x* and *y* position, and one dimension for orientation, if it is not known), it was easy and fast to perform a global search over a large region of the terrain model. This enables an even larger basin of attraction to the near-optimal solution.

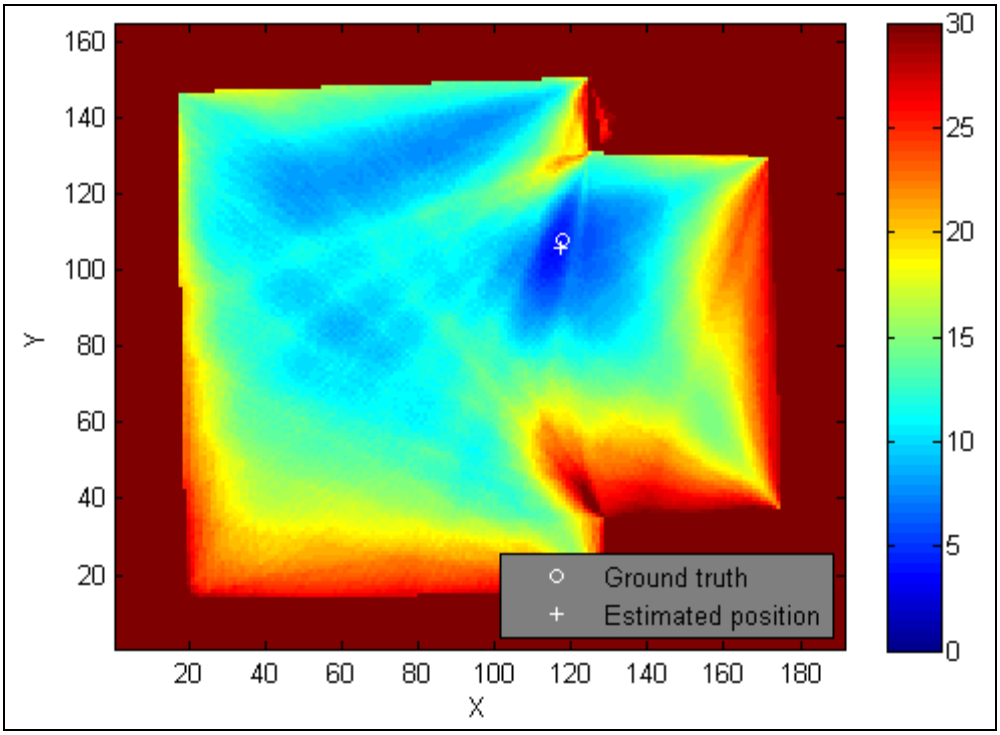

Figure 10. True and estimated camera position for ARL courtyard experiment 1. The localization error is approximately 0.5 m.

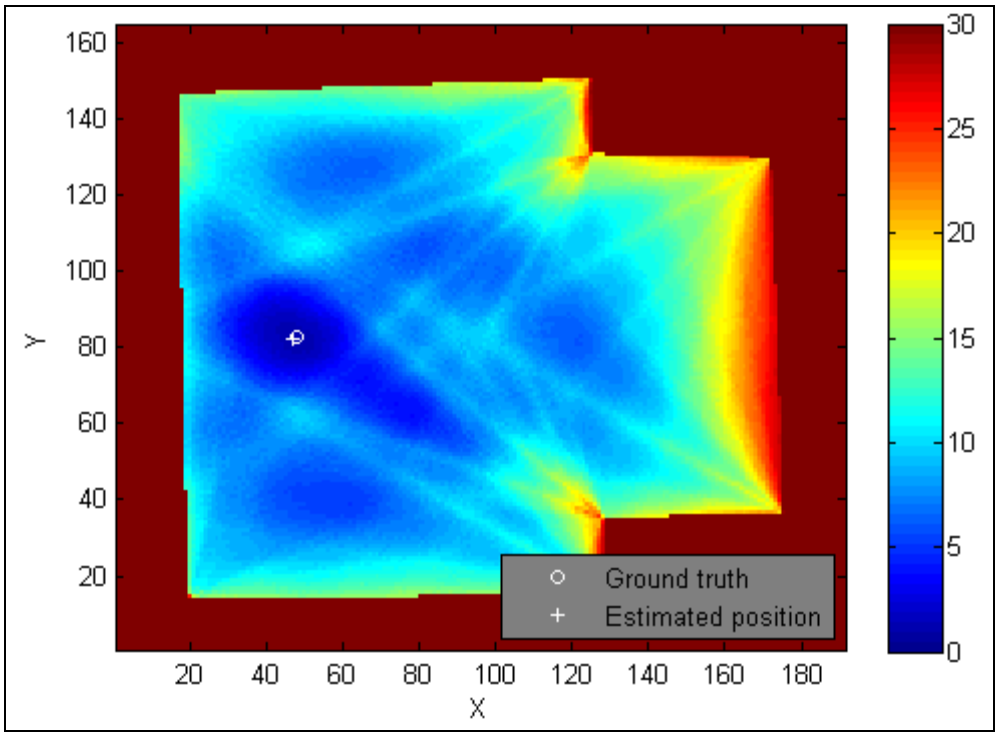

Figure 11. True and estimated camera position for ARL courtyard experiment 2. The localization error is approximately 0.5 m.

4. Conclusions

We have described an efficient algorithm to determine the position and orientation of a camera in an outdoor urban environment using camera imagery acquired from a single location on the ground. The location of the camera in the urban terrain is determined by estimating, from a single image, the footprint of visible building facades and then registering this local footprint to the terrain model. Both the local footprint estimation and the registration steps are fast. Local building footprint estimation is performed using image vanishing points to compute the 3-space orientations on the ground plane of line segments detected in an omnidirectional camera image. The local footprint orientation vector is then registered to the 2-D terrain model to determine the camera's position and orientation.

Based on initial experiments, we believe our approach is an order of magnitude more accurate than commercial GPS and it can be implemented to run in real time using modest processor resources. These qualities make the approach suitable for many applications of small platforms operating in GPS-denied urban environments such as navigation, mapping, and surveillance. Remaining work includes completing the real-time software implementation and evaluating the approach in real-world field exercises.

5. References

1. Hartley, R. I.; Zisserman, A. *Multiple View Geometry in Computer Vision*; 2nd ed., Cambridge University Press, 2004.

2. Campbell, R. J.; Flynn, P. J. A Survey of Free-Form Object Representation and Recognition Techniques. *Computer Vision and Image Understanding* **February 2001**, *81* (2), 166–210.

3. Chen, T.; Wu, K.; Yap, K.-H.; Li, Z.; Tsai, F. S. A Survey on Mobile Landmark Recognition for Information Retrieval. *Proceedings of the Int. Conf. on Mobile Data Management: Systems, Services and Middleware*, May 2009.

4. Woo, D.-M.; Nguyen, Q.-D.; Tran, Q.-D.N; Park, D.-C.; Jung, Y. K. Building Detection and Reconstruction from Aerial Images. *Proceedings of the Int. Soc. for Photogrammetry and Remote Sensing*, Beijing, China, July 2008.

5. Cord, M.; Declercq, D. Three-dimensional Building Detection and Modeling Using a Statistical Approach. *IEEE Transactions on Image Processing* **May 2001**, *10* (5), 715–723.

6. San, D. K.; Turker, M. Automatic Building Detection and Delineation from High Resolution Space Images Using Model Based Approach. *Proceedings of the ISPRS Workshop on Topographic Mapping from Space (with Special Emphasis on Small Satellites)*, Ankara, Turkey, February 2006.

7. Verma, V.; Kumar, R.; Hsu, S. 3D Building Detection and Modeling from Aerial LIDAR Data. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, Washington, DC, June, 2006.

8. Rottensteiner, F.; Trinder, J.; Clode, S.; Kubik, K. Building Detection Using LIDAR Data and Multi-spectral Images. *Proceedings of the 7th Conf. on Digital Image Computing: Techniques and Applications,* Sydney, Australia, December 2003.

9. Haithcoat, T. L.; Song, W.; Hipple, J. Building Footprint Extraction and 3-D Reconstruction from LIDAR Data. *Proceedings of the IEEE/ISPRS joint Workshop on Remote Sensing and Data Fusion over Urban Areas*, Rome, Italy, November 2001.

10. Wang, O.; Lodha, S.; Helmbold, D. P. A Bayesian Approach to Building Footprint Extraction from Aerial LIDAR Data. *Proceedings of the IEEE Third International Symposium on 3D Data Processing, Visualization and Transmission*, June 2006, pp. 192–199.

11. Müller, S.; Zaum, D. Robust Building Detection in Aerial Images. *Proceedings of the International Society for Photogrammetry and Remote Sensing Workshop CMRT: Object Extraction for 3D City Models, Road Databases and Traffic Monitoring - Concepts, Algorithms and Evaluation*, Vienna, Austria, August 2005.

12. Nevatia, R.; Lin, C.; Huertas, A. A System for Building Detection from Aerial Images. *Proceedings of the Conference on Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, Basel, Switzerland, pp. 77–86, 1997.

13. Visual Learning Systems, Inc. The LIDAR Analyst Extension for ArcGIS Automated Feature Extraction Software for Airborne LIDAR Datasets, September 2005, http://www.featureanalyst.com/lidar_analyst/publications/LA_whitepaper.pdf (accessed 2009).

14. Canny, J. A Computational Approach to Edge Detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence* **November 1986**, *8*.

15. Kovesi, P. D. MATLAB and Octave Functions for Computer Vision and Image Processing. School of Computer Science & Software Engineering, The University of Western Australia, http://www.csse.uwa.edu.au/~pk/research/matlabfns/.

16. Fischler, M. A.; Bolles, R. C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Comm. Association for Computing Machinery* **June 1981**, *24*, 381–395.

6. Transitions

We plan to transition this software to the Communications-Electronics Research Development and Engineering Center (CERDEC)/Night Vision and Electronic Sensors Directorate (NVESD) in support of the Sensor Mobility and Perception Technology Program Annex (TPA) (No. CE-CI-2008-05) and to the Safe Operations of Unmanned systems for Reconnaissance in Complex Environments (SOURCE) Army Technology Objective (ATO).

List of Symbols, Abbreviations, and Acronyms

2-D	two-dimensional
3-D	three-dimensional
ARL	U.S. Army Research Laboratory
ATO	Army Technology Objective
CAD	computer-aided design
CCD	charge-coupled device
CERDEC	Communications-Electronics Research Development and Engineering Center
DEM	digital elevation model
GPS	global positioning system
LFO	local footprint orientation
LIDAR	Light Detection and Ranging
NVESD	Night Vision and Electronic Sensors Director/Directorate
RANSAC	Random Sample Consensus
SOURCE	Safe Operations of Unmanned systems for Reconnaissance in Complex Environments
TPA	Technology Program Annex

No. of Copies	Organization
1 ELEC	ADMNSTR DEFNS TECHL INFO CTR ATTN DTIC OCP 8725 JOHN J KINGMAN RD STE 0944 FT BELVOIR VA 22060-6218
1 CD	US ARMY RSRCH LAB ATTN RDRL CIM G T LANDFRIED BLDG 4600 ABERDEEN PROVING GROUND MD 21005-5066
3 CDS	US ARMY RSRCH LAB ATTN IMNE ALC HRR MAIL & RECORDS MGMT ATTN RDRL CIM L TECHL LIB ATTN RDRL CIM P TECHL PUB ADELPHI MD 20783-1197
7 HCS	US ARMY RSRCH LAB ATTN RDRL CII A P DAVID (5 HCS) S YOUNG N FUNG ADELPHI MD 20783-1197

TOTAL: 12 (1 ELEC, 7 HCS, 4 CDS)

INTENTIONALLY LEFT BLANK.